

ALPHA

MICRO

USERS

SOCIETY

Newsletter

VOL.2 no.1

Jan., 1979

THE ALPHA MICRO USERS SOCIETY is meant to be a focal point for information about the Alpha Micro computer.

THE SOCIETIES' OBJECTIVES include:

1. Maintaining locations of service centers for the Alpha Micro computer.
2. Maintaining information about marketed software for the Alpha Micro computer.
3. Maintaining and publishing public software.
4. Publishing solutions to problems concerning the computer.
5. Responding to phone calls concerning user's problems.
6. Publishing formal requests for solutions to user needs.
7. Hold annual conventions and seminars.
8. Provide an avenue for formal requests to Alpha Microsystems concerning changes, additions, policies, enhancements, and priorities.
9. maintain information about configurations, and their special characteristics.

THE MONTHLY NEWSLETTER provides members with timely information on updates to the system, the location of known bugs, fixes to problems, examples of system features, routines, procedures, documentation, rumors, and sources for software.

The newsletter is a place for members to share frustrations, ideas, solutions, announcements, and successes.

Address correspondence and phone calls to AMUS, Jim Taylor, P.O. Box 1723, Boulder, Colorado, 80306. Phone (303) 449-8847.

Annual dues are \$25.00 per member.

AMOS version 4.0 was send out on December 20th. Several people who got copies of a pre-release reported bugs in ISAM and COPY. Alpha Micro informs us that those problems have been cleaned up for the official release. If you have the A/M business package, it was sent out with the 4.0 pre-release. You should update with the new improved 4.0.

SOME OF YOU HAVE THE NEW IMS 32K static RAM and are having trouble with the 4.0 MEMDEF command. The proper command to set up MEMDEF should read 'MEMDEF 100,377,0'. The 377 is the code to turn on all of the 32K memory. Using the configuration documented will only turn on the top 4K. Now that the 32K board works, other A/M users will want to look over the IMS boards which are cheaper than two 16K boards and consume less power. A/M has not released the 32K boards yet, watch for bigger and better things to arrive. i.e. PIICEON 64k dynamic boards.

WE GET CALLS now & again for source listings for drivers. Those should be under PPN 7,6 and you should have received the sources for all of the drivers from your dealer. If not, and your dealer doesn't have them, contact Cindy at Alpha Micro.

MICHAEL BLOCK from Box 231, Hallandale, Florida 33009 dropped us a card and asked if CP/M were a single-user or a multi-user system, and also, how the Alpha Micro rates against other micros on the benchmark programs published in KILOBAUD Magazine.

CP/M is a single user, single tasking operating system.

The benchmark programs in KILOBAUD consisted of seven programs that measured times for FOR-NEXT loops, ARITHMETIC functions, SUBROUTINE access, and IF THEN execution. We ran the programs on the Alpha Micro (no changes were necessary in the programs) in a single-user environment, and found that the Alpha Micro topped the list on most of the seven runs.

For reasons of brevity, we are listing here only the Alpha Micro times, and the timings for the first and last place machines tested by KILOBAUD. For more information consult KILOBAUD timings as reported by Tom Rugg and Phil Feldman in "Basic Timing Comparisons", KILOBAUD #10, October, 1977, pages 20-25. Timing information and benchmark programs are reproduced here with the permission of KILOBAUD magazine.

Benchmark #1

```
300 PRINT "START"  
400 FOR K = 1 TO 1000  
500 NEXT K  
700 PRINT "END"  
800 END
```

Benchmark #2

```
300 PRINT "START"  
400 K = 0  
500 K = K + 1  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END
```

Benchmark #3

```
300 PRINT "START"  
400 K = 0  
500 K = K + 1  
510 A = K / K * K + K - K  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END
```

Benchmark #4

```
300 PRINT "START"  
400 K = 0  
500 K = K + 1  
510 A = K / 2 * 3 + 4 - 5  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END
```

Benchmark #5

```
300 PRINT "START"  
400 K = 0  
500 K = K + 1  
510 A = K / 2 * 3 + 4 - 5  
520 GOSUB 820  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END  
820 RETURN
```

Benchmark #6

```
300 PRINT "START"  
400 K = 0  
430 DIM M(5)  
500 K = K + 1  
510 A = K / 2 * 3 + 4 - 5  
520 GOSUB 820  
530 FOR L = 1 TO 5  
540 NEXT L  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END  
820 RETURN
```

Benchmark #7

```
300 PRINT "START"  
400 K = 0  
430 DIM M(5)  
500 K = K + 1  
510 A = K / 2 * 3 + 4 - 5  
520 GOSUB 820  
530 FOR L = 1 TO 5  
535 M(L) = A  
540 NEXT L  
600 IF K 1000 THEN 500  
700 PRINT "END"  
800 END  
820 RETURN
```

Timings in seconds

Benchmark #	KILOBAUD 1st place	KILOBAUD last place	Alpha Micro timings
1	.9	11.5	1.21
2	4.6	39	2.88
3	8.2	92	6.33
4	9.3	110	5.65
5	10.0	121.5	6.40
6	14.8	191	12.13
7	21.6	320	23.90

KILOBAUD'S first place listing was Ohio Scientific's OSI Challenger running their 8K BASIC at 2 MHz. The CPU was a 6502 processor.

The last place was taken by IMSAI 8K BASIC version 1.3 using an 8080 processor.

The Alpha Micro timings were done via an acoustic coupler at 300 baud so we might be able to squeeze a couple of milli-seconds out of these timings running at 19.2K baud on a hardwired terminal.

DEALERS and other interested parties please note the announcement for the AMUS Novice Seminar to be held in February. We plan to create a package of materials including handouts, overhead projector displays, and other items that will be helpful in conducting a seminar for new A/M users. If you would like the materials to use in your own seminar, or would like to have a seasoned AMUS lecturer come to do a seminar for your new customers, contact us.

The following letter and program are from Francis Cox with EQUINOX Computer Systems Ltd., 32-35 Featherstone St., London, England EC1Y 8QX.

Dear Jim,

I much enjoyed the AMUS workshop in Los Angeles at the last Faire; enclosed is a simple program which you might like to publish as there was a demand from several users I met for something like it. Basically, it is like FORCE, except you can force control characters eg. ESCape, S, Q, etc. This makes it very useful for controlling batch (or background) jobs. It is also good fun controlling other people who are running something like VUE as you can make their cursor jump all over the screen; Aptly enough, the program is called POKE.

That is all for now, I will write again in a couple of months concerning a major systems software project we have under way here.

Sincerely,

Francis E. Cox

LEFFORD LOWDEN from the AM-100 User's Group has sent us a 31 page document that includes the source listings for several programs he has written for the lineprinter/spooler. They include a subroutine SPOOL.SBR that will allow BASIC users to send items to the spooler, a FORMS program that checks to see that the proper form is in the printer and allows the operator to install the proper paper in the printer before printing a file, a LPTSPL program, and PRINT. They are meant to be used in place of the A/M programs, and have several nifty features such as switches to set the form number, delete the file after printing, respond (or not respond) to form feeds, number of copies, and automatic headers and page separation.

We obviously can't reproduce all 31 pages here, but if you wish to have the programs, contact us, and we will send them to you for the price of reproducing the material. If we can find a willing volunteer, we will try to get these onto a disk.

STAN VIET hasn't been sitting on his thumbs lately. On the way back from the West Coast Computer Faire, we found an article written by him in the Continental Airlines magazine; FLIGHTTIME. Stan came up with a very interesting article on "Computers for the Family". Good work, Stan!

IN AN EFFORT to get you AMUS members to let us know what you're up to, we plan a series of questionnaires that will appear in the forthcoming issues of the NEWSLETTER. Please, fill out the answers, and send them back to us, and we will assemble the figures and publish the results. This will also help us to communicate better with Alpha Microsystems for priorities, and such, and will hopefully lead some software developers in directions where applications programs are needed.

WE HAD A GREAT TIME at the Faire, and at the Seminars that we held at the Gala Inn. Bob Currier deserves a sustained round of applause for his mammoth efforts for two days of being under the gun and answering questions from sixty eager AMUS users.

THE TRANSCRIPTS are taking time to convert to written copy from the tapes made at the seminars (mumble, mumble, nebulous reference, mumble) but we are working on them, and we will send copies to participants of the seminars as soon as they are ready.

WE WOULD LIKE TO THANK ALPHA MICROSYSTEMS for all of their help during the seminars, and especially for their generosity in providing AMUS members with a fine afternoon cocktail party on Wednesday. Many thanks also to (Len Bachus) for his talk on hardware systems, and to Bob Hitchcock and Dick Wilcox for their perseverance at the general user's meeting on Saturday, November 4th.

ACTIVITIES GENERATED AT THE SEMINARS included committees formed to study communications, word processing, and three disks full of programs. (Speaking of disks full of programs, you should have already received your disks if you asked for them. If not, we have three disks available to members for \$10.00 each. The lists of programs appears later in the newsletter. Orders may be placed with Jim Taylor.)

COMMUNICATIONS was one of the more lively subjects considered at the seminars, and a committee is looking for volunteers to set up specifications for a communication system that would allow AMUS members to contact a central computer and access updated manuals, user programs, recent AMUS notes, and any other information that AMUS members would like to have available. (Frank Yee) has offered to put his students of computer science to work on the programming if we can come up with the specs. Any volunteers? Please contact Steve Elliott at the Community Free School, PO Box 1724, Boulder, Colorado, 80306.

KHALSA COMPUTERS GENEROUSLY DONATED their communications package to AMUS at the general meeting, and those programs (source listing and all) are available on AMUS Disk #3. These programs allow you to interface you Alpha Micro to the phone lines using the D.C. Hayes 80-103A Data Communications Adapter Board. There are Three such systems working here in Boulder, and we have passed files to and from DEC-10 systems, a CDC-6400, and each other. The program TALK.PRG allows you to pass messages to other terminals on your system (or on the phone) and is a very handy program. The Free School has been using the system to send text files to a computer typesetting system, and is saving time and money thanks to Khalsa's GET.PRG and SEND.PRG.

SPECIFICATIONS ALSO are needed for a communications system that will allow transfer of information in block mode over phone lines, so that the system won't be bogged down with terminal driver interrupts.

CHUCK GAUTHIER of the ATI Data Center in Salem, Oregon wants to find a driver for the TI-810 that will utilize 1620 all of the features of that terminal. Let him know if you have such a driver. He has written the software for a Novation modem and a VT 100.

THERE IS A GREAT CRY for the sources to the floating point conversion. There are several obvious places where that information would make things a lot easier on programmers.

WE WOULD ALSO LIKE TO KNOW ABOUT drivers for the Diablo 1620 printer, and the Beehive CRT. We do know of one member using a Beehive B-100 terminal, but we don't have any information past that at the moment.

WE ARE ACCEPTING CONTRIBUTIONS of games and utility programs in source code. At least minimum documentation should be provided for each program in a file labeled file.DOC. Some members have sent in programs in object code, but we are hesitant to distribute them since they would be useless as soon as the next release appears. Please don't be shy about sending in some tiny command file that you find useful, or a short BASIC program that you have on file that could be useful to new users of an A/M system. Sometimes a program cannot be used by everyone, but it could give someone the answer to a problem they have been agonizing over for a while.

CHARLES TURNER of the No. Kato Supply Company tells us that he is having problems with his 96K system with two-
dual WANGCO drives. These include spooling from disks other than 0, converting from STD to AMS format, and building ISAM files with ISMBLD from a command file.

We have had no problems printing from disks other than 0 provided that they were labeled DSK, and not some other three character name. Converting to AMS from STD is tricky sometimes, but it does work. We have had some other reports of trouble using ISMBLD from a command file, but we have no solution right now.

Charles also suggests that A/M impliment user defined functions in BASIC, BASIC overlays, error codes for non-number input, and dynamic length strings.

SOME NOTES ON THE COMMUNICATION SEMINAR: Roger Stickney reports that a recent issue of Computer Design has information on the use of fiber optics with the RS-232 interface.

Tony Lackner has a makeshift system he wired himself to talk to a Honeywell System.

Bill Miller of Sunny Computer Systems has a SOROC CRT on a 900 foot extension using shielded cable.

Dr. Richard Sugden of Teton Data Systems is using the UDS board for phone communication.

Popular Electronics has had recent articles on the use of micro-wave transmission.

The University of Colorado Computing Center is using a very inexpensive infra-red LED system to transmit at 4800 baud over a several mile range.

The A/M system now ignores parity which can be a problem in communications. Francis Cox is currently rewriting the micro code for the parity checking.

IN AN EFFORT TO IMPROVE AMUS'S ability to get disks, newsletters, and help out to members, we are moving to an office in downtown Boulder, and are going to share a secretary with a local lawyer. This means that someone will be there to answer the phone during the day, mail will get collected daily, and disks and other materials will get the prompt attention they deserve. The Post Office Box will remain the same, and we will publish the phone number after it is installed on January 31st.

A M U S S E M I N A R
February 15 & 16, 1979
For NOVICE Alpha Microsystems Users
Held in Boulder, Colorado
(Skiers take note)

The Folks from Data Processing Consulting, Inc. will be conducting a two day "Novice only" seminar February 15th and 16th, 1979. There will be four half-day sessions consisting of:

1. How to bring up your system:
SYSTEM.INI
Initialization programs
Memory management
AMS vs STD format
2. BASIC::
File formats
BASORT
ISAM
MAPs
Other tricks we've learned
Questions and answers
3. EDIT, VUE, FORMAT
Other AMOS programs
Command files
4. Use of Assembly XCALL routines
Communications
Questions and answers on anything
Time to play with the machine

The \$150 charge will include the two days of seminars, materials, machine time, and lunch. Sessions start at 9:00 AM both mornings.

We will make room and transportation arrangements for you, and also skiing arrangements if you like.

The class will be limited to 10 prepaid participants. Additional classes will be scheduled as interest demands.

Contact Jim Taylor at Data Processing Consulting, Inc.
PO Box 1723, Boulder, Colorado 80306 (303) 449-8847.

Amus disk #1

DSK1: (1,4)

AMUS01 LST 4

DSK1: (11,13)

ABAGEL	BAS	5	ACEYDU	BAS	5	AMAZIN	BAS	9	ANIMAL	BAS	5
ANIMAL	DAT	2	AWARI	BAS	5	BAGEL1	BAS	5	BAGEL2	BAS	5
BANDIT	BAS	4	BANG	BAS	2	BANNER	BAS	6	BATTLE	BAS	9
BIO	BAS	6	BOAT	BAS	9	BOUNC1	BAS	3	BOUNC2	BAS	3
BOWL	BAS	6	BOXNG1	BAS	8	BOXNG2	BAS	7	BUTTON	BAS	5
CALNDR	BAS	3	CHANGE	BAS	3	CHECKR	BAS	6	CHECK1	BAS	12
COMBAT	BAS	7	COMP	BAK	5	COMP	CMD	5	COMPOS	BAS	3
DANGLE	BAS	3	DATING	BAS	7	DAY	BAS	2	DEFUSE	BAS	4
DEPTH	BAS	3	DICE1	BAS	2	DICE2	BAS	2	DIMND2	BAS	2
ENGLIS	DAT	3	FIPFOP	BAS	4	FLIGHT	BAS	13	FRENCH	DAT	1
FURS	BAS	12	GOMOKO	BAS	5	GUNER1	BAS	5	GUNNER	BAS	4
GYPSY	BAS	6	HAMURA	BAS	5	HANG1	BAS	8	HANG2	BAS	8
HANG3	BAS	7	HILO	BAS	3	HIQ	BAS	8	HORSE	BAS	3
HURKLE	BAS	6	LUNAR1	BAS	7	MADRL1	BAS	5	MANDLA	BAS	3
MASTER	BAS	6	MUGWMP	BAS	4	MUSICA	BAS	1	MUSICB	BAS	2
MUSICC	BAS	2	MUSICD	BAS	1	MUSICE	BAS	1	PATERN	BAS	1
PICTUR	BAS	1	QUBIC	BAS	13	RVRSE1	BAS	4	RVRSE2	BAS	4
SCIF11	BAS	11	SCIF12	BAS	11	SNARK	BAS	5	SPLAT	BAS	11
STATES	BAS	9	STOCK	BAS	10	SWAMI	BAS	8	TAXMAN	BAS	9
TCTAC1	BAS	10	TCTAC2	BAS	6	TCTAC3	BAS	7	TUTR01	BAS	9
TUTR02	BAS	6	TUTR03	BAS	9	TUTR04	BAS	6	WUMPUS	BAS	12
WUMP2	BAS	12	WUMP3	BAS	9	23	BAS	3	LIST	LST	3
LIST4	BAK	3	LIST	CMD	1	LIST4	LST	3	LIST	BAK	1

Total of 92 files in 508 blocks

Grand total of 93 files in 512 disk blocks

AMUS disk #2

DK1:(1,4)AMUS02 LST 3 SYSINI HOG 2 SYSTEM INI 3
Total of 3 files in 8 blocks

DSK1:(2,2)
HOG CMD 4 PIG CMD 4 TREK CMD 6
Total of 3 files in 14 blocks

DSK1:(11,12)
ASCI11 LST 13 ASCII2 LST 13 BASWRD LST 5 BPRINT BAS 2
DOFW BAS 2 ECHO MAC 1 KOMON1 MAC 1 KOMON2 MAC 5
OUTPUT BAS 7 PACKER BAS 17 RESEQ1 BAS 26 RESEQ2 BAS 6
RESEQ3 BAS 6 SINGLE MAC 1 SLEEP MAC 1 TXTABR LST 3
TXTPRT BAS 5
Total of 17 files in 114 blocks

DSK1:(11,13)
ADDS1 BAS 5 ADDS2 BAS 6 CHECK BAS 12 CHESS1 BAS 26
COUNT1 BAS 5 COUNT2 BAS 6 CUBE BAS 10 DIMND1 BAS 2
GRAPH BAS 17 GRAPH LST 1 JOKE DAT 6 KING BAS 18
LUNAR1 BAS 6 SOLIT BAS 33 SPACWR BAS 24 STAR1 BAS 19
STAR2 BAS 20 STRTRK BAS 23 STRTRK HLP 5 WLDAPWR BAS 14
Total of 20 files in 258 blocks

DSK1:(11,14)
BKSP MAC 5 DATE MAC 16 FBKSP MAC 4 IRS BAS 7
PACKHX MAC 6 PRAD50 MAC 7 PRIME DAT 1 PRIME1 BAS 1
PRIME2 BAS 2 PRIME3 BAS 2 PRTARG MAC 1 PRTVAR MAC 2
PYMD MAC 9 STRTST MAC 1 UNPKHX MAC 6 URAD50 MAC 2
USERNO MAC 3 UYMD MAC 6
Total of 18 files in 81 blocks

Grand total of 61 files in 475 disk blocks

AMUS disk #3

DSK1:(1,4)

CALL	PRG 2	COUNT	PRG 1	DETACH	PRG 1	DIRLST	PRG 1
DSKFIX	PRG 3	GET	PRG 2	GET3	PRG 1	GIVE1	LST 3
HANGUP	PRG 1	INIT1	CMD 1	INIT2	CMD 2	LD	PRG 1
SEND	PRG 1	SYSTEM	INI 2	TALK	PRG 1	TALKTO	PRG 1
TALK3	PRG 1	AMUS01	DIR 4				

Total of 18 files in 29 blocks

DSK1:(1,6)

DCH IDV 1

DSK1:(2,2)

FF	CMD 1	FF	DAT 1	HELLO	CMD 1	HELP	CMD 3
POLITE	CMD 2	REMOTE	PRM 2	RES	CMD 2	SECURE	CMD 1
STATUS	CMD 1						

Total of 9 files in 14 blocks

DSK1:(11,11)

AMUS LST 2 AMUS03 LST 2

Total of 2 files in 4 blocks

DSK1:(11,12)

ASCII	BAS 3	ATLNTA	GA 3	BOMBER	BAS 6	CALL	MAC 10
CHICAGO	IL 3	DCH	DOC 38	DCH	MAC 11	DCH	TXT 33
DETACH	DOC 2	DETACH	MAC 4	DIRLST	DOC 2	DIRLST	MAC 7
DIVIDE	BAS 1	DSKDDT	DOC 4	DSKFIX	DOC 7	DSKFIX	MAC 9
FTBL	BAS 13	GET	MAC 9	GET3	MAC 7	GPSK	CMD 2
HANGUP	MAC 2	HIQ	BAS 13	HURKLE	BAS 4	LOGS	LST 4
LOGS	MAC 2	LOGS	PRG 1	LPTSPL	CLR 8	LPTSPL	FNL 9
LPTSPL	MAC 7	LPTSPL	ORG 6	MESSAG	BAS 2	MESSAG	CMD 1
NICOMA	BAS 2	PARAB	BAS 3	PIGRAF	BAS 1	PRINT	CLR 7
PRINT	FNL 8	PRINT	MAC 7	PRINT	ORG 6	QDT	DOC 4
ROACH	LTR 8	RPLFRM	BAS 3	SEND	MAC 6	SNTCLR	CA 3
SYSTEM	INI 2	SYSTEM	IN2 2	TALK	DOC 2	TALK	MAC 4
TALKTO	MAC 8	TALK3	MAC 6	TAXMAN	BAS 9	TAYLOR	LTR 11
TEMP1	LST 11	TEMP2	LST 12	TEMP3	LST 13	TEMP4	LST 11
TEMP5	LST 12	TEMP6	LST 12	TOF	LST 1	TYPE	BAS 14
TYPRED	BAS 2						

Total of 61 files in 423 blocks

DSK1:(11,14)

WEKDAY BAS 9

DSK1:(11,15)

IFTHEN	LST 5	INTRO	LST 13	KIDS1	LST 3	KIDS2	LST 2
KIDS3	LST 2	KIDS4	LST 2	KIDS5	LST 2	KIDS6	LST 1
KIDS7	LST 1	README	LST 10	STRING	LST 6		

Total of 11 files in 47 blocks

Grand total of 103 files in 527 disk blocks

AUTHOR DISTRIBUTED PROGRAMS

NEED TO EXCHANGE DATA BETWEEN THE OUTSIDE WORLD AND YOUR ALPHA-MICRO?

We have a package to allow you to exchange data via the IBM 3747 series machines and the Alpha-Micro.

Capabilities:

1. Go from 9-track 800 bpi tape to floppy disks via the IBM 3747 (or equivalent) then to your Alpha-Micro.
2. Dump data or program files from your AMOS structured devices to an IBM 3747 compatible floppy disk and then to tape.

How it works: Three assembly language routines called from a user BASIC program form the heart of the system. These allow you to open a IBM compatible floppy for input/output, read off a file to an AMOS file, or write a file of your choice onto the floppy. ASCII to EBCDIC conversions are performed automatically for you. These subroutines can be used in conjunction with a custom BASIC program to fit almost any needs.

-Other programs in the Package:

OUT.BAS	Sample general utility program for writing files from AMOS to IBM 3747 format.
INPUT.BAS	Sample general utility program for reading files from IBM 3747 format to an AMOS file.
IMGDMP.BAS	Utility program used to examine data on an IBM 3747 floppy.
IMGHEX.BAS	Utility program used to examine data on an IBM 3747 floppy in hex without EBCDIC to ASCII conversion.
DUMP.BAS	Utility program to examine a random AMOS file.
?????.CMD	Sample command files to use the above programs and by which to model your own.

COST: \$400.00

TO ORDER OR FOR ADDITIONAL INFORMATION CONTACT:

Eugene C. Platt
10333 Northwest Freeway
#518
Houston, Texas 77092
713-681-6949

ALPHA MICRO ADDRESS RETENTION

We developed AMAR in April for our own use. The Data Bank does a lot of mailing list maintenance for accountants and bookkeeping services and we needed a program that would allow us to manipulate many small mailing lists quickly and efficiently utilizing only one floppy disk.

AMAR stores first name, last name, 2 address lines, city, state, zip, telephone number and an optional comment field for each record. The input data is edited for length and format prior to storage and can be easily retrieved on the CRT or in printed format. There are twenty programs in the system, giving the user the flexibility to do just about anything he wants with the data stored. The user may access any file (or any record in any file) on request; AMAR monitors the number of files on the disk and the number of records in each file and keeps the user apprised of these totals. Galley sheets (name and address rosters) may be printed or displayed on the CRT at any time and mailing labels may be printed in 3, 4, or 5 line format. The sort module allows the user to sort on any field in the record (up to 3 sort keys) or he may optionally choose to sort in any one of 7 standard formats. The entire system is menu-oriented with CRT formatting features that make AMAR very easy to use.

We are making the object code available (on floppy) to AM-100 users for \$50. This includes the user's manual, which ordinarily may be ordered for \$4.00.

The source code is available to AM-100 dealers only, on a limited license basis.

Contact Tom Herod, The Data Bank, 70 West Palmetto Drive, Pasadena, CA 91105. Phone: 213-795-8611.

INNOVATIVE COMPUTING

7446 4th Avenue Northeast, Seattle, Washington 98115

Telephone: (206) 525-7788

**** N O W A V A I L A B L E ****

A SOFTWARE UTILITY THAT ALLOWS THE AM-100* OWNER ACCESS TO THE VAST AMOUNT OF SOFTWARE, BOTH PUBLIC AND PRIVATE DOMAIN, NOW AVAILABLE ON CP/M* DISKETTES. THE UTILITY PERFORMS THE FOLLOWING FUNCTIONS:

- 1) DISPLAY A CP/M DIRECTORY
- 2) SPOOL A CP/M DIRECTORY TO THE LINE PRINTER
- 3) COPY A CP/M ASCII FILE, BY NAME, TO AN AMOS FILE
- 4) COPY AN AMOS ASCII FILE, BY NAME, TO A CP/M FILE
- 5) DISPLAY THE CONTENTS OF A CP/M FILE ON THE CONSOLE DEVICE
- 6) SPOOL A DSKANA /L TYPE DUMP OF A CP/M DISKETTE

THE CP/M USERS GROUP CURRENTLY HAS IN EXCESS OF 30 DISKETTES FULL OF PUBLIC DOMAIN SOFTWARE RANGING FROM GAMES TO THE COMPLETE SOURCE FOR BUD SHAMBURGER'S GENERAL LEDGER PUBLISHED IN INTERFACE AGE. THE UTILITY, CONSISTING OF A BASIC PROGRAM AND AN ASSEMBLER SUPPORT ROUTINE IS AVAILABLE 'OFF THE SHELF' IN AMS OR IBM FORMAT.

PRICING: RUN-TIME MODULES ONLY	\$30.00
RUN-TIME MODULES AND ALL SOURCE CODE	\$50.00

CONTACT: JOH. SCHALLER
INNOVATIVE COMPUTING
11300 3rd NE #216
SEATTLE, WA 98125
(206) 362-8111

- * CP/M is a trademark of DIGITAL RESEARCH
- * AM-100 is a trademark of ALPHA MICROSYSTEMS



Specialists in Micro Computer Business Systems

Dec. 7, 1978

AMUS
Attn: Jim Taylor
P.O. Box 1723
Boulder, Colorado 80306

Dear Sir:

Enclosed is the instruction manual for RATBAS (rational basic) which I mentioned in our phone conversation. Also enclosed are a number of examples of Ratbas code, and it's output which is standard basic code.

We would appreciate if you could publish the first page of the manual and the BNF definition of the lanuage. (space permitting)

It is our feeling here at Micobus that writing structured code can save the programmer about 40% of his or her time and increase the reliability and readability of his or her software product.

Thank you,

Irvin M. Shapiro
Director Software Development

IM:sm

PS: We appreciate whatever you are able to publish, If you have any questions about Ratbas please feel free to call me at (312) 676-0966. We will be selling Ratbas for \$350.00, this includes the source written in Ratbas.

RATBAS USERS GUIDE
RELEASE 1.0

WRITTEN BY: IRVIN M. SHAPIRO
MICROBUS, INC.
7855 GROSS PT. RD.
SKOKIE, IL. 60077
(312) 676-8966

COPYRIGHT 1978 MICROBUS, INC.

INTRODUCTION:

A MAJOR PROBLEM WITH THE BASIC PROGRAMING LANGUAGE IS ITS LACK OF STRUCTURED PROGRAM SEQUENCE CONTROLS. THIS FORCES THE PROGRAMMER TO USE A VAST NUMBER OF GOTO STATEMENTS WHICH DECREASE THE READABILITY OF THEIR CODE. IN TURN THE WRITTING OF UNSTRUCTURED CODE LEADS TO PROGRAMS WHICH ARE HARD TO DEBUG AND EVEN HARDER TO MODIFY.

RATBAS PROVIDES THE USER WITH PROGRAM STRUCTURES WHICH ALLOW HIM (OR HER) TO WRITE STRUCTURED MODULAR CODE, WITHOUT THE HIGH OVERHEAD OF A LARGER ALPHABASIC RUN TIME SYSTEM. THE ADVANTAGES OF RATBAS INCLUDE SMALLER PROGRAMS, LESS DEBUGGING TIME, AND MORE EASILY READ CODE.

OPERATION:

RATBAS IS A PREPROCESSOR WHICH ACTS AS A FIRST PASS TO THE STANDARD ALPHABASIC COMPILER. THE OUTPUT OF THE RATBAS PREPPROCESSER IS A STANDARD ALPHABASIC PROGRAM. ALL LEGAL ALPHABASIC STATEMENTS ARE ALSO LEGAL RATBAS STATEMENTS. THE ONLY RESTRICTION ON THE ALPHABASIC PROGRAMMER IS THAT LINE NUMBERS MAY NOT BE USED IN RATBAS PROGRAMS. THIS MAY ACTUALLY BE THOUGHT OF AS AN ADVANTAGE SINCE IT FORCES THE USER TO USE ONLY LABELED GOTOS AND GOSUBS. (MORE ABOUT GOTOS LATER) IN ADDITION TO THE ALREADY ADVANCED FEATURES FOUND IN ALPHABASIC, RATBAS ADDS 3 NEW PROGRAM STRUCTURES TO THE LANGAUGE. THESE ARE DO LOOPS, BLOCK STRUCTURED IF THEN ELSE STATEMENTS, AND EASY TO USE SINGLE STATEMENT ISAM FILE CONTROL STRUCTURES. THE NEXT SECTION DESCRIBES THESE NEW FEATURES IN A MODIFIED FORM OF BNF.

STATEMENT SYNTAX:

NOTE: <> SYMBOLS ARE USED TO SIGNAFY OPTIONAL KEYWORDS. ALL RATBAS STATEMENT DELIMITERS MUST BE THE FIRST EXECUTABLE KEYWORDS ON A LINE. THESE KEYWORDS MAY BE POCEDED BY COMBINATIONS OF TABS AND SPACES. AND I HIGHLY RECOMEND THAT TABS AND SPACES BE USED TO FORMAT YOUR CODE INTO A MORE READABLE FORM.

WHEN AN INDEX IS SPECIFIED THE DATA RECORD IS NOT UPDATED. ONLY THE SECONDARY KEY FILE GETS CHANGED.

UPDATE #NUM, KEY = STR, <ERROR = LABEL, >IOLIST

USED TO UPDATE AN EXISTING RECORD IN A FILE

DELETE #NUM, KEY = STR, <INDEX = NUM, ><ERROR = LABEL, >IOLIST

WHEN AN INDEX IS SPECIFIED ONLY THAT INDEX IS DELETED, THE DATA RECORD IS NOT ALTERED.

GENERAL USE INSTRUCTIONS:

THE RECOMMENDED WAY TO WRITE RATBAS PROGRAMS IS TO USE VUE TO CREATE A PROGRAM. THEN TO RUN THIS PROGRAM THROUGH RATBAS AND THE COMPILER. YOU WILL FIND A COMMAND FILE ON YOUR DISK WHICH WILL DO THIS FOR YOU IN ONE SIMPLE STEP. TO USE THIS COMMAND FILE JUST TYPE RATBAS THE COMPUTER WILL THEN PROMPT YOU FOR THE NAME OF THE PROGRAM YOU WANT TO COMPILE. THE DEFAULT EXTENSION ON THIS PROGRAM IS RAT. AFTER LOCATING YOUR PROGRAM THE COMPUTER WILL SEARCH FOR A PROGRAM WITH THE SAME NAME AS YOURS AND AN EXTENSION OF BAS. IF IT FINDS SUCH A PROGRAM IT WILL ASK YOU IF YOU WANT TO DELETE IT. AN ANSWER OF Y WILL CAUSE IT TO DELETE THE CURRENT BAS PROGRAM AND WRITE OUT THE RATBAS TRANSLATION TO A FILE WITH YOUR PROGRAM NAME AND AN EXTENSION OF BAS. IF YOU RELY IN THE RATBAS OUTPUT WILL GO TO A FILE WITH THE EXTENSION OF RBS.

TO WRITE ACCURATE AND EFFICIENT CODE IT IS IMPORTANT TO UNDERSTAND A LITTLE ABOUT HOW RATBAS WORKS. THERE ARE 3 KEYWORDS WHICH ARE USED IN CONJUNCTION WITH DO LOOPS WHICH MAY CAUSE THE USER SOME CONFUSION.

DO WHILE CONDITION THIS STRUCTURE CHECKS THE WHILE CONDITION AT THE TOP OF EVERY LOOP. IT DOES NOT CHECK THE CONDITION AFTER EVERY STATEMENT. THEREFORE THE PROGRAMMER MUST STRUCTURE HIS PROGRAMS SO THAT ANY CODE IN THE MIDDLE OF A LOOP DOES NOT HAVE TO CAUSE A BRANCH TO THE END OF THE LOOP. THE LOOP WILL CONTINUE TO EXECUTE UNTIL THE WHILE CONDITION IS FALSE AT THE TOP OF THE LOOP.

DO UNTIL CONDITION THIS STRUCTURE CHECKS THE CONDITION AT THE BOTTOM OF THE LOOP. THE LOOP WILL CONTINUE TO EXECUTE UNTIL THE CONDITION IS TRUE AT THE BOTTOM OF THE LOOP. PLEASE NOTE THAT THIS STRUCTURE TRANSFERS CONTROL ON A TRUE CONDITION AND THE WHILE STRUCTURE TRANSFERS CONTROL ON A FALSE CONDITION.

THIS THEN BRINGS UP THE QUESTION OF WHAT TO DO IN THE CASE WHERE A DECISION TO EXIT A LOOP MUST BE MADE IN THE MIDDLE OF THAT LOOP. AN EXAMPLE WOULD BE WHEN YOU HAVE AN INPUT STATEMENT IN A LOOP AND YOU

WANT TO EXIT ON AN END OF FILE CONDITION.

EXIT ON CONDITION.....EACH TIME EXECUTION PASSES THIS TEST A CHECK IS MADE TO SEE IF THE CONDITION IS TRUE. IF THE CONDITION IS TRUE THE PROGRAM CONTROL IS IMMEDIATELY TRANSFERED TO THE END OF THE LOOP.

WITH THE ADDITION OF THE RATBAS PROGRAM STRUCTURES IT IS NOW POSSIBLE AND HIGHLY ADVANTAGEOUS TO WRITE STRUCTURED CODE. IN SOME CIRCLES THIS IS CALLED GOTO LESS CODE. I BELIEVE HOWEVER THAT THERE ARE CERTAIN TIMES WHEN THE LIMITED USE OF GOTOS MAY MAKE YOUR CODE EASIER TO WRITE AND MORE READABLE.

A GENERAL PROGRAMING TECHNIQUE THAT I HAVE USED WITH RATBAS THAT SEEMS TO WORK VERY WELL, IS TO WRITE YOUR PROGRAMS AS A NUMBER OF SMALL MODULES, WHICH MAY BE COMPILED SEPERATELY. WHEN ALL THE MODULES ARE DEBUGGED, MOVE ALL MAP STATEMENTS TO THE BEGINNING OF THE MAIN PROCEDURE AND USE THE APPEND UTILITY TO "LINK" YOU MODULES TOGETHER. THEN RECOMPILE THE FINAL PROGRAM AS A SINGLE MODULE.

RATBAS LIMITATIONS:

THE PRESENT VERSION OF RATBAS WAS WRITTEN IN RATBAS BY USING A BOOTSTRAPING TECHNIQUE. THEREFOR SINCE THE FINAL EXECUTABLE CODE IS AN ALPHABASIC RUN MODULE IT IS NOT AS FAST AS IT COULD BE. RATBAS WILL TRANSLATE FROM 100 TO 150 LINES A MINUTE DEPENDING ON THE TYPE OF DISK YOU ARE USING AND THE NUMBER OF USERS ON YOUR SYSTEM. A FUTURE CURENTLY UNSCHEDULED RELEASE OF RATBAS WILL IMPROVE ITS RATE OF THROUGHPUT BY USING A NUMBER OF ASSEMBLER SUBROUTINE CALLS. THE CURRENT VERSION OF RATBAS REQUIRES A MERMORY PARTITION SIZE OF ABOUT 15K TO RUN CORRECTLY. SINCE IT IS A DISK BASED PREPROCESSER IT WILL TRANSLATE A PROGRAM OF ANY SIZE (THAT WILL FIT ON THE DISK) IN THIS AMOUNT OF MEMORY.

ANY OF THE RATBAS PROGRAM STRUCTURES MAY BE NESIED UP TO 10 LEVELS DEEP. IF THE USER ATTEMPS TO NEST ANY OF THE STRUCTURES TO MORE THEN 10 LEVELS THE TRANSLATION PROCESS WILL BE TERMINATED AT THAT POINT. ALSO PLEASE NOTE THAT THE USER IS NOW LIMITED TO HAVING 25 INDEXED FILES OPEN AT ANY ONE TIME. (I DO NOT CONSIDER THIS A LIMITATION, SINCE I WOULDN'T WANT TO TRY TO KEEP TRACK OF MORE THEN 25 OPEN FILES AT ONCE)

THE RATBAS PREPROCESSOR KEEPS TRACK OF ALL OPEN INDEXED FILES SO THAT IT CAN TELL WHICH STATEMENTS IN A PROGRAM ARE RATBAS ISAM STATEMENTS. IN ORDER TO KEEP THE SIZE OF RATBAS DOWN, IN WAS NECESSARY TO PUT A

RESTRICTION ON THE USE OF THE CLOSE STATEMENT. THE CLOSE STATEMENT FOR AN ISAM FILE MUST BE PLACED SEQUENTIALLY BELOW ALL ISAM I/O STATEMENTS IN THAT PROGRAM. BY SEQUENTIALLY I MEAN PHYSICALLY PAST THE I/O STATEMENTS IN THE TEXT FILE WHICH CONTAINS THE RATBAS CODE. A SIMPLE WAY OF DOING THIS IS TO CLOSE ALL FILES IN SUBROUTINES THAT ARE AT THE END OF THE CODE.

PROGRAM EXAMPLES:

ON YOUR SOURCE DISK YOU WILL FIND 3 PROGRAMING EXAMPLES. THE FIRST EXAMPLE CALLED TEST.RAT IS A PROGRAM THAT DOES NOTHING, BUT IS USED AS AN EXAMPLE OF STATEMENT SYNTAX. THE SECOND PROGRAM IS THE ISAM EXAMPLE PROGRAM NOW CALLED ISMTST.RAT THAT ALPHA MICRO PROVIDED, MODIFIED TO USE THE RATBAS ISAM STATEMENTS. AND THE THIRD PROGRAM IS THE ACTUAL RATBAS PREPROCESSOR WRITTEN IN RATBAS. THIS FINAL EXAMPLE IS A FAIRLY GOOD EXAMPLE OF STRUCTURED CODE USING RATBAS.



Linktec Consultancy Limited

40, HING FAT ST., 3/FL., B6
CAUSEWAY BAY, HONG KONG.

TELEPHONE : 5-783592

November, 15, 1978.

Alpha Micro Software

BASXRF.PRG this program written in assembler reads the BASIC source file and produces a cross reference. Its command format is,

BASXRF oflnam.XRF=iflnam.BAS

default file extensions are .XRF and .BAS respectively. An intermediate file with extension .IXF is used to hold the cross reference pointers.

```

; Memory Test Program for AM-100 Timesharing System
; Version 2.0 - Revised December 3, 1978
; Copyright 1978 by Richard N. Rubinstein, M.D.
; 1021 Broadview Rd., Ft. Washington, PA 19022
; This source file is called.MCH:TST.MAC
;
; This program will test all locations above and below itself,
; including the system monitor area, by verifying that all zeroes
; and all ones can be written and read. It will not interfere with the
; System Monitor or with jobs being run at the same time by other users.
;
; It will not test non-contiguous memory such as VDM (if non-contiguous)
; since the program will automatically stop when it ascends to any word
; which cannot be changed from all ones, as would be the case for non-
; existent or non-functional memory.
;
; The program will print the location and description of any errors. It will
; run continuously, starting again from 0 when it reaches the end of
; RAM memory. When Control-C is typed the program will terminate and print
; the size of contiguous RAM present in the system. If no errors were en-
; countered, this will be explicitly stated.
;
; It is recommended that a priority of 1 (low) be assigned to the job
; running this program.
;
; The following are System Macro-instructions copied from SYS.MAC:
;
; CRLF - Prints carriage return and line feed on console
; TYPECR - Prints message followed by CRLF on console
; TYPESP - Prints message followed by space on console
; LOCK - Disables interrupts
; UNLOCK - Enables interrupts
; DCVT - Prints contents of R1 (in decimal) on console
; CTRLC - Branches to address specified if Control-C typed
; TTY - Sends the byte in R1 to the console
; EXIT - Terminates execution and returns control to command level
;
; COPY SYS/P ; Include system macro definitions into this program
;
; Set initial conditions:
;
; INTRO: NOP ; This NOP is interpreted by System Monitor as permission
; ; to run this program for users who are not logged in
;
; CRLF
; CRLF
; MOV #12.,R1
; TTY ; Send this BYTE to console - will clear screen if CRT
; TYPECR Memory Test Program - Copyright 1978 by R. N. Rubinstein, M.D.
; TYPECR This program will test all contiguous RAM above and below itself
; CRLF
; MOV #65534.,R3 ; Initially assume memory size of 64K
; CLR R4 ; Flag is 0 until the first error is found
;
;
; R0 will hold the exclusive upper boundary of the region being tested.
; R1 will hold the address of the memory location being tested.
; R2 will temporarily store the contents of the location being tested.
; R3 will hold the (exclusive) upper bound of contiguous memory.
; R4 will be used as a flag to signal whether errors have been found.
;
;
; Start of main program loop:
;

```

```

MAIN: CLR R1 ; Testing to start at memory location 0
      LEA R0,INTRO ; Test all locations up to but not including start of program
      CALL TEST
      LEA R1,LAST ; Begin testing again after bypassing this program
      MOV R3,R0 ; This time test until end of entire contiguous memory
      CALL TEST
      BR MAIN ; Repeat entire process
;
; This is the subroutine which performs the actual memory test
;
TEST: MOV @R1,R2 ; Save contents of location to be tested
      CMP R1,R0 ; Have we reached the end of the region being tested?
      BNE MORE ; If not, continue
      RTN
MORE: LOCK ; Disable interrupts until memory contents are restored
      CLR @R1 ; Store zeroes
      TST @R1 ; See if location now contains all zeroes as expected
      BEQ NOERR1
      CMP #177777,@R1 ; See if all ones
      BNE NEXT
      MOV R1,R3 ; Since it contains all ones, this location is absent or kaput
      RTN ; So this is the effective upper boundary of contiguous memory
NEXT: TYPEPSP Cannot store zeroes at ; Type error message on console
      DCVT 0,3 ; Send decimal address to console
      CRLF
      SET R4
NOERR1: SET @R1 ; Now try to set to all ones
      CMP #177777,@R1 ; Did this work?
      BEQ NOERR2 ; Yes
      TYPEPSP Cannot store ones at ; Type error message on terminal
      DCVT 0,3 ; Send decimal address to terminal
      CRLF
      SET R4
NOERR2: MOV R2,@R1 ; Restore contents of memory location
      UNLOCK ; Enable interrupts so other users can get their licks
      CTRLC FINIS ; Wrap things up if Control-C has been typed in
      IW2 R1 ; Increment R1 by 2 to address next word
      BR TEST ; Get set to test next word
;
; Now wrap things up:
;
FINIS: TST R4 ; If error flag is not set, announce the purity of memory
      BNE DONE ; Otherwise keep quiet
      TYPEPSP Nc errors were found!
DONE: TYPEPSP Memory size is
      MOV R3,R1
      DCVT 0,3 ; Announce amount of working contiguous memory (in decimal)
      TYPEPSP < Bytes.>
;
; The next two instructions clear a Control-C flag in the System
; Monitor Job Control Block.
;
MOV @#JOB CUR,R1
BIC #J.CCC,@R1
;
CRLF
EXIT
;
LAST: ; First memory address after end of program
;
END

```

KHALSA

COMPUTER SYSTEMS INCORPORATED



500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 684-3311

1978 NOV 29

Dear Jim :

Here is buglist #8, having to do with the AMOS system released with the 1.0 Accounting package. It is correctly referred to as pre-4.0 release, but I refer to it as 3.5, with no loss of clarity.

Please make a note in the next newsletter that the COMMON.MAC that was printed in AMUS 8/9 has one bug : the single BGE mnemonic should be changed to BHI (it is right in the middle of the page). The version being distributed on the AMUS library diskettes has this bug already corrected.

Very happy to have met you at the AMUS convention. Please keep up the good work; we (and the world) need more dedicated people like yourself.

Yours truly,

Bob Fowler

Bob Fowler



COMPUTER SYSTEMS INCORPORATED



500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 684-3311

AMOS BUGS (Version "3.5") continued

- (6) RUN --- no file name causes crash

Typing RUN at monitor level with no file name following causes system to go dead.

- (7) DIR --- wrong byte-size in some cases

The output from the following monitor-level commands :

DIR MEM:

DIR RES:

gives incorrect sizes for the programs involved. The correct values are equal to :

(value displayed by AMOS) - 10 bytes

- (8) SORT --- two bugs

In all cases that I have tried, SORT adds 2 blank lines to the end of the sorted file. This even occurs with the smallest non-trivial case of all, i.e., sort a file containing the single-character line "A". What specifically happens is that

CR-LF-CR-LF

is appended to the end. A second bug occurs when the file is too big to sort in memory at one time. In this case, the final merging phase fails to occur, and the temporary sort files

SRTMP1.TMP , SRTMP2.TMP , etc.

are left on disk. [Mike Roach says the latter bug has been fixed].

- (9) Command files --- escape characters sometimes get hung up

Sometimes it is clever to edit a file via a command file. This requires escapes (ASCII 27) to be sent AMOS from the command file. This appears to hang up the system (sometimes !). Sometimes, this hangup can be cleared by manually hitting a carriage return; sometimes the hangup cannot be cleared. One user reports that he was able to get around this problem by inserting extraneous carriage returns in his command file, BEFORE the escapes that are likely to hang up the system.

- (10) VUE --- disables HOG option on SIO boards

There are other known bugs with VUE, but Duane Cowgill reports that if a crt is brought with an IMSIO.IDV driver using the HOG option, and VUE is used and exited, then the HOG option is no longer in effect (type TRMDEF to make sure). Thereafter, the screen output of VUE is sloooooooow, unless the system is re-initialized.



COMPUTER SYSTEMS INCORPORATED

One Kap Sat Nam Karta Puran Nih Bhao Nih Ver Aral Mochet Auni Sai Bhang Gur Prasad, Jap. Ad Sach Jugad Sach Heli Sach Nanar Hosi Bi Sach

500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 684-3311

AMOS SUGGESTIONS (Version "3.5") 11/29/78

(1) BAUD --- allow no argument

In keeping with the spirit of many of the other AMOS utilities, may I suggest that BAUD (without any argument) answer with the user's current baud rate, rather than an error message ("baud rate 0 is illegal")

(2) DSKFIL --- allow random files

As of now, DSKFiling a random file usually leads to a dead-end because AMOS looks for next-block pointers (as if the file were a sequential file). It would be nice if an error message "Cannot DSKFIL a random file" were displayed. It would be very nice if DSKFIL simply checked the file-type and displayed the blocks correctly, for all files.

(3) FORTRAN --- overlays please

For all of us with stacks of ANSI FORTRAN code, just waiting for ALPHA-FORTRAN, please please implement ANSI. If any bells/whistles are included beyond ANSI, then please give overlay capabilities some serious consideration.

(4) TXTFMT --- document justify command

For AMOS people mainly, new commands in TXTFMT are /JUSTIFY or /J /NO JUSTIFY or /NJ The first command enables simultaneous right-left justification. Extra blanks are "stuffed in" to make the text line up on both left and right sides. The second command disables this option.

(5) ALPHABASIC --- new file input syntax

The following is a legitimate input from a random file : READ #1, XDATA[A,B] Where #1 refers to a previously-opened random file, and XDATA could be a string variable (or anything, but watch out !). With this syntax, all files may now be ALLOCATED, OPENed, READ, and WRITtEn using the same statements.

One Kap Sat Nam Karta Puran Nih Bhao Nih Ver Aral Mochet Auni Sai Bhang Gur Prasad, Jap. Ad Sach Jugad Sach Heli Sach Nanar Hosi Bi Sach

One Kap Sat Nam Karta Puran Nih Bhao Nih Ver Aral Mochet Auni Sai Bhang Gur Prasad, Jap. Ad Sach Jugad Sach Heli Sach Nanar Hosi Bi Sach



COMPUTER SYSTEMS INCORPORATED



DR. KAN SAT NAM KARTA PURKH NIB BHAL NIB VER ANAL MOONT AJUR SAI BHANG GUR PRASAD JAP AD SACH J. AL SACH HEBI SACH NANAK HOM BI SACH

500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 664-3311

AMOS SUGGESTIONS (Version "3.5")

11/29/78

(6) ALPHABASIC --- document certain F6 variables better

Certain kinds of variables must be in F6 format to operate correctly. They are : FOR loop counters, LOOKUP result variables, and OPEN record number variables. This is properly documented under "MAP" on page 20, but should be redundantly documented under the sections on FOR, LOOKUP, and OPEN. Some users go through long phone calls before figuring this out.

(7) ALPHABASIC --- document precise algorithm for numeric input

ALPHABASIC doesn't give errors on any inputs, it simply takes as much as it can & goes on. In some cases, this is nice, e.g.

11/29/78

11,29,78

23:30:30

are all legitimate inputs to the statment

INPUT A,B,C

The input algorithm for numeric inputs is as follows :

(a) starting from the left end, use as many characters as possible to form a numeric input until an illegitimate character is found. If the very first character is an illegitimate character, then the input number defaults to a value of zero.

(b) The first illegitimate character is considered a delimiter and is passed over.

(c) the next input starts at the character following the delimiter.

The input algorithm for string inputs is as follows :

(a) starting from the left end, if the first character is a double quote, then the input consists of all characters between this double quote & the next double quote (if a second double quote isnt found, then the input ends at the carriage return). The next input starts with the character following the second double quote (or following the carriage return); if this first character is a comma, it is passed over

(b) If the first character is not a double quote, then the input consists of all characters up to the first comma (if a comma isnt found, then the input ends at the carriage return). The next input starts with the character following the comma (or the carriage return).

DR. KAN SAT NAM KARTA PURKH NIB BHAL NIB VER ANAL MOONT AJUR SAI BHANG GUR PRASAD JAP AD SACH J. AL SACH HEBI SACH NANAK HOM BI SACH

DR. KAN SAT NAM KARTA PURKH NIB BHAL NIB VER ANAL MOONT AJUR SAI BHANG GUR PRASAD JAP AD SACH J. AL SACH HEBI SACH NANAK HOM BI SACH

DR. KAN SAT NAM KARTA PURKH NIB BHAL NIB VER ANAL MOONT AJUR SAI BHANG GUR PRASAD JAP AD SACH J. AL SACH HEBI SACH NANAK HOM BI SACH



COMPUTER SYSTEMS INCORPORATED



PAK SAT NAM KARTA PURAN NIR BHAG NIR VER ANAL MOORT AJUNI SAI BHANG GUR PRASAD JAP AD SACH JUGAD SACH HEDI SACH NANAK HOBI BI SACH

500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 684-3311

AMOS SUGGESTIONS (Version "3.5")

11/29/78

(8) ALPHABASIC --- document full potential of CHAIN

The AMOS manual (page 68) states that CHAIN may be used to transfer control either to another .RUN module, a .PRG module, or a .CMD command file. However, that is not all there is to life. It turns out that, with one minor exception that may be easily circumvented, the CHAIN command may contain an entire command file within the CHAIN command itself.

Consider the following program segment :

```
MAP1 COMMAND,S,300
MAP1 CRLF ,S,2
CRLF=CHR(13)+CHR(10) ! carriage return + line feed
COMMAND="DSK0:TIME.PRG[1,4]" + CRLF + ":T" + CRLF + "SORT FRED.DAT"
COMMAND=COMMAND+CRLF+"10" + CRLF + "5" + CRLF + "1" + CRLF + "A" + CRLF
COMMAND=COMMAND+" " + CRLF + "RUN FRED"
```

The above, in essence, is equivalent to the following command file :

```
TIME
:T
SORT FRED.DAT
10
5
1
A
```

RUN FRED

Several things are demonstrated by this sample :

- (a) the first line may not be a ":T" or similiar command but the remaining lines may be one of these.
- (b) like a command file, all outputs are not displayed on the crt unless ":T" or ":R" are used. Normally the user would prefer "silence". If he desires to see all outputs, however, he must put a "dummy" command in the first line (e.g., "TIME" above), because this line cannot be made to print out.
- (c) the program could return to itself at the end of the CHAIN. In the above example, RUN FRED does this.
- (d) there are no external program modules besides FRED.RUN and the usual AMOS utilities (and the temporary data file FRED.DAT). Several kinds of command file sequences like the above could be contained within the same BASIC program or (better) manufactured by the program itself before CHAINing. This would be somewhat like having command files with parameters and branching.

PAK SAT NAM KARTA PURAN NIR BHAG NIR VER ANAL MOORT AJUNI SAI BHANG GUR PRASAD JAP AD SACH JUGAD SACH HEDI SACH NANAK HOBI BI SACH



COMPUTER SYSTEMS INCORPORATED



500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101 TELEPHONE (213) 684-3311

AMOS NOTES & FREEBIES

11/29/78

- (1) BADDSK.LST --- see enclosed

For people who find that they have disk I/O errors, especially ERROR 10 and ERROR 20, this 2-page write-up can give them a good idea of where they stand and what they should do. It covers several possibilities of what may be wrong, and describes a half-dozen AMOS utilities that may be of value.

- (2) MOUNT2.MAC --- see enclosed

Not to be confused with Alpha Micro's MOUNT.SBR (which prompts the user to physically mount certain devices), MOUNT2 is a donated by Mike Franks and actually mounts a device via an ALPHABASIC XCALL. You must macro-assemble MOUNT2 and then rename the .PRG module to .SBR, as with all subroutines.

- (3) DSKDDT and QDT documentation --- see enclosed

These are not currently documented in the AMOS manual. These 2 pages of documentation are part of the Khalsa AM-100 Telecommunications package.

- (4) TXTFMT summary --- see enclosed

Courtesy of Guruprem Singh Khalsa. A handy, one-page synopsis of all the text format commands.

KHALSA COMPUTER SYSTEMS INCORPORATED
 500 SOUTH LAKE AVENUE, PASADENA, CALIFORNIA 91101
 TELEPHONE (213) 684-3311
 FAX (213) 684-3311
 Telex: 153121 KHALSAL
 Cable: KHALSA
 BITE BIT

Fixing Crunched AM-100 Diskettes

If you think that your diskette may have some bad (unreadable) blocks on it, then first type the following :

```
SET DSKERR  
REDALL DSKn:
```

All bad blocks will be reported. To see which blocks are contained in which files, type :

```
DSKANA DSKn:/L
```

Unfortunately, DSKANA will exit after hitting the first bad block encountered. To do a DSKANA on one file, type :

```
DSKFIL filespec
```

In any of the above programs (REDALL, DSKANA, DSKFIL), AMOS will try to read the diskette block 8 times; after that it either goes on to the next block (REDALL) or ends the program. If AMOS can read a block within the 8 tries allowed, then your chances of saving it without typing it over are good. Otherwise, you may have to say goodbye to it. The following sections give several field-tested tips on how to salvage as many of these bad blocks as possible, in the easiest way.

To Fix Individual Bad Blocks

First of all, for each bad block, write down its file name, disk number and block number. Try fixing the block by typing

```
DSKDDT DSKn:<block #>
```

The same disk read errors will be displayed as before, but DSKDDT will not automatically exit. It has loaded the block into user memory and is waiting for a command. If the user now types the single character

```
E (no carriage return)
```

then DSKDDT will write the block back to the disk. In the process of doing this, it will also correct the block "hash total". If you are lucky, the block is now fixed. Test it by running DSKFIL or DSKDDT again. If DSKDDT does not fix the bad block, then the block will have to be re-formatted, which can only be done by re-formatting the entire diskette. The user must retrieve everything he can off the bad diskette, copying it to a good disk, then FORMAT the bad disk. Unfortunately, the program DSKCPY does not accomplish this task. DSKCPY blindly copies everything (good blocks, bad blocks, unused blocks), and (if it doesn't bomb out first) will only create another bad disk. Unfortunately also, ERASEing only the bad file off the bad disk will not solve the problem, because the bad block is now unused (but still bad !) and will show up again as soon as it is allocated to another file. You've heard the worst ! Read on for some helpful hints.

Cleaning Up the Bad Disk

The best way to clean up the bad diskette is as follows :

1. Salvage as much of the bad file(s) as possible. Various tricks may be of use. For example, for a BASIC source code file (filnam.BAS), type :

```
BASIC
```

```
LOAD filnam
```

BASIC will stop loading when the bad block is encountered, but you can now save the first part of the file by typing :

```
SAVE filnam2
```

where filnam2 should be a different file name from any other file. Whatever tricks you try, do not ERASE any bad files (thereby liberating the bad blocks) until you are through playing around on the bad disk.

2. ERASE the bad files, and run DSKANA once more to see if the remaining files on the bad disk are ok.

3. Copy the remaining (good) files onto another diskette. Whether the second diskette is the system disk, a backup disk, a scratch diskette, or a new diskette, you should first run a REDALL on this second diskette to determine if it is good or bad. If this second diskette is bad, then it should either be "cleaned up" after the current cleanup, or else FORMATED into oblivion, and then REDALLed again. If (after being FORMATED) the second diskette is still bad (especially if it is brand new), then there are probably disk drive hardware problems. If you have Version 4.0 of AMOS (or Version 1.0 of the Alpha Accounting System), you can copy all the files onto the good disk by typing :

```
LOG 1,2
```

```
COPY DSKgood[:] = DSKbad[:] / NODELETE
```

This will copy all files by the same name, into the same account numbers, create the account numbers on the good diskette (if they don't already exist), and will not copy over a file by the same name (the user will have to decide what to do with these files after the COPY). If you have an earlier AMOS version, you must create any new accounts yourself (using SYSACT), and for each account number, type :

```
LOG ppn
```

```
COPY /X = DSKbad:*
```

4. Finally, after everything possible has been saved off the bad disk, type :

```
FORMAT
```

```
<bad disk drive #>
```

and FORMAT those bad blocks away. Now do a REDALL on the diskette. If there are still bad blocks on this diskette, then it is either physically damaged (a candidate for the circular file), or else there is trouble with the disk drive hardware. If no other diskettes seem to be having problems, then it is probably the "bad" diskette itself which was at fault. Look at the oxide surface of the diskette for scratches (block 0 is on the outer edge, block 763 (IBM format) or 1147 (AMS format) is at the inner edge). If all diskettes are having similar read/write problems, then it is probably the hardware (or system software configuration) at fault.

Disk Fix Program

If there is a bad block in any directory, then the user will not be able to access some or all of the files in that directory. If the Master File Directory (block 1) is crunched, no directories whatever on the bad disk will be accessible. In these cases and in particularly messy cases in general, a last resort is the very powerful program

```
DSKFIX.PRG
```

It was written by Larry White (of Caltech) and is available from Khalsa Computer Systems. Inquiries welcome.

Backups

On all computer systems, and especially on a floppy disk system, the user should always keep himself covered with backups. If the user has a backup, then instead of going through all of the above, all he has to do is FORMAT the bad diskette, do a REDALL to see if it is physically beyond repair, and then do a DSKCOPY from the backup diskette to the newly-formatted diskette.

NAME

DSKDDT - disk editor

USAGE

DSKDDT DSKm:nnn
m = disk number
nnn = block number (octal)

DESCRIPTION

DSKDDT allows the user to inspect and change disk. Operation is similar to DDT and QDT, but the commands are slightly different.

DSKDDT reads the specified block into memory and allows the user to edit the words of the block. On exit (E) the block is written back to disk.

E Exit from DSKDDT. Write block to disk.
 Forget about the entry in progress.
Z Set all words in the block to zero.
O Set all words in the block to 177777.

The following commands are preceded by an octal number:

/ Display specified location. No number entered is treated as if zero was entered.
<CR> Carriage Return stores the specified number in the current location. No change is made if no number was entered.
<LF> Line Feed works like <CR> but also goes on to the next location and displays its contents.
^ Up-arrow or Circumflex is just like <LF> but decrements the current location.
N N sets the block number so that the modified block is written back to disk at another location.

COMMENTS

This program is very powerful. Control-C is useful when you've wiped out the block in memory and don't want it written to disk.

PROGRAMMER

The people at Alpha Microsystems.

BUGS

Lower case letters are not recognized.

On exit from DSKDDT to the operating system, AMOS does not completely restore the terminal mode. If you type something and hit CTRL-U, you will need to hit CTRL-C before going on.

Addresses outside the size of the block are permitted.

NAME

QDT - memory editor

USAGE

QDT

DESCRIPTION

QDT allows the user to inspect and modify memory on an AM-100. Operation is similar to DDT, such that each command to QDT is a single keypress. However, QDT can change anywhere in memory but has fewer commands.

QDT keeps track of a "current location" in memory and a "current mode" of display. The following are the commands of QDT, and a brief description of what they do.

E Exit from QDT.
B Change to BYTE mode.
 In byte mode, single bytes are modified.
 The current location changes by one.
W Change to WORD mode.
 In word mode, full words are modified.
 The current location changes by two.
Display current word as two octal bytes.
<TAB> TAB takes the contents of the current location and uses
 the contents as the next location.
 Rubout or Delete will wipe out all portions of the
 command or number currently being entered.

The following commands may be preceeded by an octal number.
If no number is entered, the specified location is not changed.

/ Slash changes the current location to the specified
 number and displays the new location.
<CR> Carriage Return stores the specified number in the
 current location.
<LF> Line Feed works as <CR> and advances to the next location.
^ Up-arrow or Circumflex is just like <LF> but decrements
 the current location.

COMMENTS

This program is very powerful. You can easily wipe out the operating system with it, even if you know what you are doing.

QDT is useful for testing devices on the S-100 bus. I/O ports start at 177400 in memory, and proceed up to 177777. One can read and write a single port at a time by entering Byte mode.

PROGRAMMER

The people at Alpha Microsystems.

BUGS

Lower case letters are not recognized.

On exit from QDT to the operating system, AMOS does not completely restore the terminal mode. If you type something and hit CTRL-U, you will need to hit CTRL-C before going on.

Text Formatting:

```
/BR          = /BREAK
/I n         = /INDENT n
/C text      = /CENTER text
/L n         = /LINE n
/LS          = /LIST
/LSAL        = /LIST ALPHA LOWER
/LSAU        = /LIST ALPHA UPPER
/LSRL        = /LIST ROMAN LOWER
/LSRU        = /LIST ROMAN UPPER
/LE          = /LIST ELEMENT
/ELS         = /END LIST
/EB          = /ENABLE BAR
/DB          = /DISABLE BAR
/BB          = /BEGIN BAR
/EB          = /END BAR
/BOLD        = /BOLD
/NO BOLD     = /NO BOLD
```

Page Formatting:

```
/CHAP text  = /CHAPTER text
/APP text   = /APPENDIX text
/HL n text  = /HEADER LEVEL n text
/HP         = /HEADER PAGE
/HNP        = /HEADER NO PAGE
/HB         = /HEADER BOLD
/HNB        = /HEADER NO BOLD
/NM n       = /NUMBER n
/NNM        = /NO NUMBER
/NMABS n    = /NUMBER ABSOLUTE n
/NMB n      = /NUMBER BOTTOM n
/NNMB       = /NUMBER NO BOTTOM
/NMRL n     = /NUMBER ROMAN LOWER n
/NMRU n     = /NUMBER ROMAN UPPER n
/TTL text   = /TITLE text
/NTTL       = /NOTITLE
/P n        = /PAGE n
/CT text    = /CENTERED TITLE text
/NCT        = /NO CENTERED TITLE
```

Mode Setting:

```
/F          = /FORMAT
/U          = /UNFORMAT
/J          = /JUSTIFY
/NJ         = /NO JUSTIFY
```

Parameter Setting:

```
/X n        = /LINESIZE n
/Y n        = /PAGESIZE n
/M n        = /MARGIN n
/S          = /SINGLE
/D          = /DOUBLE
```

software by
dravac

DRAVAC (DRAV-ack) LTD is committed to producing and selling the highest quality product. The software systems described here are our current and soon-to-be-released products. They are programmed to run on the Alpha Microsystems AM-100* computer.

While a relatively new company, we are rich in experience. The average programming experience of our technical staff is over seven years. We are by no means newcomers to the field of computer software. Our experience varies from designing and implementing major banking systems to developing environmental watchdog programming systems for solar heated/cooled structures. We will undertake custom programming projects at our prevailing rates.

QUALITY SOFTWARE FOR YOUR

ALPHA *
MICROSYSTEMS™ computer

WORD PROCESSING

Maximum power combined with ease of use are the primary characteristics of our word processing system. Simple two-letter commands imbedded in the input text control all aspects of print formatting. Justification, centering, headings, page numbers, etc., are all handled automatically. In addition, more sophisticated features are included, such as creating a table of contents, merging a mailing list into a form letter -- in all over sixty commands, covering just about any type of word processing function you can think of!

Two levels of our word processing system are available: the basic system and the advanced system. The basic system can be upgraded to the advanced level through the purchase of additional modules.

THE BASIC SYSTEM:

A complete word processing system in itself, the basic system consists of the modules WORD and SEX(TM). Using only these two programs the user can prepare text documents and then format them.

WORD is the text processor program. This module converts the input text file into the final output format, by executing the commands imbedded in the input text.

SEX™ is our line number oriented text editing system. It can be used not only to prepare text files for WORD, but also to produce program source files or what have you. Using dynamic paging techniques, SEX can be used to edit a 64-kilobyte file with a partition size only 1.5 k larger than the SEX program. Featuring a full range of commands (change, move, copy, insert, delete, etc.), SEX boasts a sophisticated command structure which permits great flexibility. For example, the command "LIST 'John' IN COLUMNS 1/20 IN 1,3,5/10 UNN" will list on the terminal all lines containing the string "John" anywhere in columns 1 through 20 of lines 1, 3, and 5 through 10, omitting the line numbers. Most SEX commands allow this type of structure.

THE ADVANCED SYSTEM:

Encompassing all the features of the basic word processing system, the advanced system also allows the user to: generate KeyWord In Context indices (KWIC); generate standard format indices; verify spelling via a self-updating spelling program; and generate glossaries.

The programs needed to perform these functions are also available separately.

SOFTWARE TOOLBOX™

To develop good software, you need good software tools. The Alpha Microsystems computer comes with the basic set of tools. DRAVAC LTD offers you the SOFTWARE TOOLBOX(tm), a collection of programming aids designed to cut system development and implementation time while helping you produce a more reliable product.

UTILITIES include a disk PATCH program which allows you to read, modify and write any sector on a disk; a set of "XCALL" routines for ALPHABASIC* programs; and BASIC utilities.

C LANGUAGE developed by Dennis Ritchie at Bell Telephone Laboratories, is an extremely versatile structured programming language. "C" offers a unique combination of brevity and readability due to its extremely clear control structure. DRAVAC LTD has designed certain extensions to the basic "C" language, including interfaces to AMOS* routines, to afford the user the optimum combination of hardware and software.

TINY-C is a stripped-down version of the "C" language, which is interpreted rather than compiled. DRAVAC LTD has been licensed by tiny-C Associates to produce the AM-100 implementation of tiny-C. We have made several enhancements during the course of this implementation. Both "C" and tiny-C are steadily growing in popularity because of their extreme ease of use coupled with lack of restrictions. Many of DRAVAC LTD's future programming projects will be carried out entirely in "C".

ANSI COBOL is currently under development for release first quarter 1979. DRAVAC COBOL will support all level 1 features and all level 2 verbs. In addition, many extensions in the areas of string handling and CRT screen formatting will be available, as well as an interface to our DBMS system. ISAM files will be supported.

GAMES is a collection of nine (or more) games, including such favorites as OTHELLO, HAMMURABI, KING and WIZARD. After all, all work and no play just might make your Alpha start dropping bits!

DATABASE MANAGEMENT

THE ELECTRIC MEMO:

PDQS is DRAVAC LTD's simple database management system, which allows the user to start using his computer in a meaningful fashion within minutes of setting up. PDQS utilizes pseudo-natural language (any language!) for input and retrieval of data. The database may consist of from one to 36 datasets, each storing up to 1750 items of information. An item of data can be retrieved by specifying any part of the record.

DBMS is currently under development for first quarter 1979 release. Our DBMS package is a standalone system providing hierarchial access to the data using associative logic. DRAVAC COBOL will provide an interface to this system, thus allowing programs written in this extremely popular business language easy access to the data.

UPDATE SERVICE:

Most of DRAVAC LTD's major program packages are supported by an optional update service. All enhancements and new releases will be forwarded to purchasers of annual service contracts. This ensures that you will always have the most recent and powerful version of the product as soon as it is available.

DISTRIBUTED BY:



dravac ltd
13 east 30 street
new york, ny 10016
212-889-1556

* AMOS, ALPHABASIC, AM-100, ALPHA MICROSYSTEMS are all trademarks of Alpha Microsystems .
SEX and SOFTWARE TOOLBOX are trademarks of Dravac Ltd.

SOFTWARE PAC #1

"SOME COMMON BASIC PROGRAMS"

This package consists of 76 Programs which appeared in Osborne & Associates book, "Some Common Basic Programs," which was written by Lon Poole and Mary Borchers. The programs are broken up into four different areas:

- (1) Statistics
- (2) Finance
- (3) Math
- (4) Miscellaneous

This package is useful to many individual needs for it covers a very broad range of applications. On the following page is a list of all the programs which are included in this package.

The programs were originally written in Wang Basic, but we have converted these basic programs to the individual basic languages that we have listed below for sale. All packages include one or more floppy disks, a supplementary instruction manual and may also include the above mentioned book, "Some Common Basic Programs," which does serve as the users manual. This book provides listings of all of the programs, example runs and information on how to alter the programs if you desire to change it slightly for a particular need.

ALPHA-MICRO ALPHA BASIC	/ 1 DISK	/ \$15.00 WITHOUT BOOK	/ \$23.50 WITH BOOK
CP/M — CBASIC	/ 2 DISKS	/ \$22.00 WITHOUT BOOK	/ \$30.50 WITH BOOK
CP/M — MICROSOFT BASIC	/ 2 DISKS	/ \$22.00 WITHOUT BOOK	/ \$30.50 WITH BOOK
TRS-80 LEVEL II BASIC	/ 2 DISKS	/ \$22.00 WITHOUT BOOK	/ \$30.50 WITH BOOK
APPLE — APPLESOFT II	/ 2 DISKS	/ \$22.00 WITHOUT BOOK	/ \$30.50 WITH BOOK

ALL ORDERS MUST BE PREPAID AND WE PAY ALL POSTAGE.

BUSINESS
SOFTWARE



MICRO-COMPUTER
SYSTEMS CONSULTING

THE BASIC BUSINESS SOFTWARE COMPANY, INC.
POST OFFICE BOX 2032
SALT LAKE CITY
UTAH 84110
(801) 363-1199

The following programs are included in Software Package Number One:

FUTURE VALUE OF AN INVESTMENT	REAL ROOTS OF POLYNOMIALS: NEWTON
FUTURE VALUE OF REGULAR DEPOSITS (ANNUITY)	ROOTS OF POLYNOMIALS: HALF-INTERVAL SEARCH
REGULAR DEPOSITS	TRIG POLYNOMIAL
REGULAR WITHDRAWALS FROM AN INVESTMENT	SIMULTANEOUS EQUATIONS
INITIAL INVESTMENT	LINEAR PROGRAMMING
MINIMUM INVESTMENT FOR WITHDRAWALS	MATRIX ADDITION, SUBTRACTION, SCALAR
NOMINAL INTEREST RATE ON INVESTMENTS	MULTIPLICATION
EFFECTIVE INTEREST RATE ON INVESTMENTS	MATRIX MULTIPLICATION
EARNED INTEREST TABLE	MATRIX INVERSION
DEPRECIATION RATE	PERMUTATIONS AND COMBINATIONS
DEPRECIATION AMOUNT	MANN-WHITNEY U TEST
SALVAGE VALUE	MEAN, VARIANCE, STANDARD DEVIATION
DISCOUNT COMMERCIAL PAPER	GEOMETRIC MEAN AND DEVIATION
PRINCIPAL ON A LOAN	BINOMIAL DISTRIBUTION
REGULAR PAYMENT ON A LOAN	POISSON DISTRIBUTION
LAST PAYMENT ON A LOAN	NORMAL DISTRIBUTION
REMAINING BALANCE ON A LOAN	CHI-SQUARE DISTRIBUTION
TERM OF A LOAN	CHI-SQUARE TEST
ANNUAL INTEREST RATE ON A LOAN	STUDENTS T-DISTRIBUTION
MORTGAGE AMORTIZATION TABLE	STUDENTS T-DISTRIBUTION TEST
GREATEST COMMON DENOMINATOR	F-DISTRIBUTION
PRIME FACTORS OF INTEGERS	LINEAR CORRELATION COEFFICIENT
AREA OF A POLYGON	LINEAR REGRESSION
PARTS OF A TRIANGLE	MULTIPLE LINEAR REGRESSION
ANALYSIS OF TWO VECTORS	NTH ORDER REGRESSION
OPERATIONS ON TWO VECTORS	GEOMETRIC REGRESSION
ANGLE CONVERSION: RADIAN TO DEGREE	EXPONENTIAL REGRESSION
ANGLE CONVERSION: DEGREE TO RADIAN	SYSTEM RELIABILITY
COORDINATE CONVERSION	AVERAGE GROWTH RATE FUTURE PROJECTIONS
COORDINATE PLOT	FEDERAL WITHHOLDING TAXES
PLOT OF POLAR EQUATION	TAX DEPRECIATION SCHEDULE
PLOT OF FUNCTIONS	CHECK WRITER
LINEAR INTERPOLATION	RECIPE COST
CURVILINEAR INTERPOLATION	MAP CHECK
INTEGRATION: SIMPSONS RULE	DAY OF THE WEEK
INTEGRATION: TRAPEZOIDAL RULE	DAYS BETWEEN TWO DATES
INTEGRATION: GAUSSIAN QUADRATURE	ANGLO TO METRIC
DERIVATIVE	ALPHABETIZE
ROOTS OF QUADRATIC EQUATIONS	

SOFTWARE PAC #2

"ALPHA-MICRO UTILITIES"

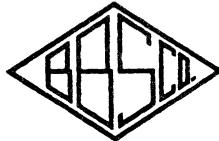
The Basic Business Software Company, Inc. is proud to announce the availability of a package of utility programs for the Alpha-Micro user. The following programs are included on one floppy disc, and a 53-page user's manual.

- 1) **SETTIM** — used in setting the real-time clock in the AM-100, and also setting the current date.
- 2) **TIMDAY** — will display the date and time of day. May be used as a subroutine in a program to extract date and time of day for use.
- 3) **LIST** — will spool an Alpha-Basic file(s) to your assigned line printer. It will number all pages, and print the date and program name in expand print on the first line of each page. It will also print your name or company name and time on the first page of the listing. Will space each page so that there is 3 blank lines at the top and bottom of each page.
- 4) **XREF** — will cross reference a basic file. It will list a variable, constant or reserved word, and print all lines that it is used in.
- 5) **LETTER** — will allow you to quickly write a letter. It allows input of margins, spacing, etc. and will format the letter into a file for the printing of the letter.
- 6) **ISAM** — consists of 10 programs which allows the user to use the Alpha-Basic ISAM statements with little trouble. The user needs only to load these programs with the main program and use a statement such as GOSUB 'ISAM 'NEW 'RECORD to add a record to the ISAM file.
- 7) **CRTBLD** — will let the user build a CRT screen image, and allow editing. It will then store it as a file for the user to call during the program, or it will add line numbers and print statements to use it as a subroutine.
- 8) **CRTSUB** — This program is quite amazing, and will save the programmer many hours of work. It allows the programmer to enter all of the X, Y coordinates where a "?" should appear, the input variable name, and the length of the input. It will then write a subroutine in Alpha-Basic which will contain all of the BASIC code including error checking. The programmer need only add a gosub to perform the entire inputting of a CRT screen of information.
- 9) **CREATE** — allows the user to enter the number of logical records, and the length of each. The program will then calculate the number of physical records to save. It also allows the user to try different record lengths to get the best use of the disk space.

- 10) **SFTORF** — allows the user to convert a sequential file to a random file.
- 11) **RFTOSF** — allows the user to convert a random file to sequential.
- 12) **RFTORF** — allows the user to convert a random file to another random file which may be larger or smaller.
- 13) **QUERY** — allows the user to enter a file name, and it will tell you if that file is a random or sequential, and will print the number of records, the largest record length, and the total size of the file.
- 14) **SORT** — will sort a random file or sequential file using BASORT, will allow the user to enter all of the variables from the terminal.
- 15) **LOAD** — will load in modules from DSKO: [1, 4] which allow the user to perform disc copies, copying from one disk to another without the systems disk in drive zero.

PRICE: \$25.00, Postage Paid
Includes 1 disk and 53-page manual

*BUSINESS
SOFTWARE*



*MICRO-COMPLTER
SYSTEMS CONSULTING*

THE BASIC BUSINESS SOFTWARE COMPANY
POST OFFICE BOX 2032
SALT LAKE CITY
UTAH 84110

Software Pac #3

STATISTICS

The Following Statistic programs are included,
and all will run in 8K of memory:

- | | |
|---------------------------------|-------------------------------------|
| 1. Permutations + Combinations | 13. T-Distribution |
| 2. Basic Statistics | 14. Chi-Square Evaluation |
| 3. Moments, Skewness & Kurtosis | 15. Chi-Square Distribution |
| 4. Anova (one-way) | 16. F-Distribution |
| 5. Anova (two-way) | 17. Bivariate Normal Distribution |
| 6. Linear Regression | 18. Geometric Distribution |
| 7. Exponential Curve Fit | 19. Poisson Distribution |
| 8. Logarithmic Curve Fit | 20. Binomial Distribution |
| 9. Power Curve Fit | 21. Weibull Distribution |
| 10. Multiple Linear Regression | 22. Spearman's Rank |
| 11. Normal Distribution | 23. Contingency Table |
| 12. T-Statistics Evaluation | 24. Logarithmic Normal Distribution |

Available on: Alpha-Micro/TRS-80/CBASIC/MBASIC/Apple

Price: \$30.00 includes disk and manual

*BUSINESS
SOFTWARE*



*MICRO-COMPUTER
SYSTEMS CONSULTING*

THE BASIC BUSINESS SOFTWARE COMPANY, INC.

POST OFFICE BOX 2032

SALT LAKE CITY

UTAH 84110

(801) 363-1199

Software Pac #4 "FINANCE CALCULATOR"

This package consists of two programs which will perform most of the functions offered in financial calculators...

FINANCE

Lump Sums

- Present Value
- Future Value
- Interest Rate
- Number of Periods

Bonds

- Coupon Rate
- Discount Rate
- Yield to Maturity
- Purchase Price

Annuity

- Payment at Beginning
- Payment at End
- Balloon Payments
- Future Value
- Present Value
- Payment Amount
- Interest Rate
- Number of Periods

R.O.R.

This program will calculate the rate of return on an investment given either a group of uneven cash flows or groups of even cash flows.

It will calculate the following:

1. Net Present Value
2. Internal Rate of Return
3. Adjusted I.R.R.

Price \$15.00 includes disk or cassette and manual
Available: Alpha-Micro/TRS-80/CBASIC/MBASIC
APPLE/PET/SORCERER

BUSINESS
SOFTWARE



MICRO-COMPUTER
SYSTEMS CONSULTING

THE BASIC BUSINESS SOFTWARE COMPANY, INC.
POST OFFICE BOX 2032
SALT LAKE CITY
UTAH 84110
(801) 363-1199

FARM COMPUTERS, INC.
1010 SUNSET COURT
WEST LAFAYETTE, IND. 47906

November 22, 1978

Mr. Jim Taylor
P. O. Box 1723
Boulder, Colorado
80306

Dear Mr. Taylor:

Farm Computers, Inc. is a recently formed company that is involved in developing specialized software for agriculture. We discovered quite early a critical need to write an editor to facilitate software and documentation development. The editor is now complete, and we have a text formatter that is nearing the final stages of completion. The editor is complete; the documentation is complete; and we thought others would want to know of a quick and easy editor!

The FCI editor is EASY to learn! No cryptic keystrokes or counting characters to jump over in FCI-ED. The editor is a line-oriented editor, with the ability to change a single character in a line -- the type in use on many mini and midi systems today. Ten different editing commands are provided -- 9 are single-letter commands. All are recognized in upper or lower case. The editor proper, of course, is upper/lower case for textual work. The editor can be used to copy or move sections of text, or create new files which are partials or composites of other files.

Lots of documentation is available for FCI-ED -- written in a non-technical, straight-forward manner! Secretaries can read it, learn it, and use it -- with a minimum of supervision. Over 21 single-spaced pages, with running examples, a 2-page reference guide, and even 5 pages of "AMOS for Beginners".

The FCI editor, which should run on any terminal, model 33 up to HP-2648A, is available as follows:

- Option #1 -- FCI-ED Demo diskette.* \$10.00 Includes full copy of the documentation and floppy. Demo version allows experimenting with all editor commands and features, except two. The \$10.00 credits toward purchase with options #2 or #3.
- Option #2 -- FCI-ED Editor diskette.* \$75.00 Includes full copy of the documentation and floppy. Full editor features ready for use with Alpha-BASIC 3.4 -- free floppy exchange when 4.0 released.
- Option #3 -- FCI-ED Editor diskette.* Same as Option #2, except for use with Alpha-BASIC 4.0 -- WHEN AVAILABLE!

* ALL floppies sent as PERSCI 277 AMS Format.

Address inquiries and requests to --

ATTN: TSM/ED
Farm Computers, Inc.
1010 Sunset Court
West Lafayette, IN 47906



MANAGEMENT MONITORS, INC.
DATA PROCESSING

17908 LA SALLE AVENUE
GARDENA, CALIFORNIA 90248
321-8487 • 324-8555

November 22, 1978

Alpha Micro Users Society
& Community Free School
P.O. Box 1724
Boulder, Colo. 80306

Gentlemen:

We purchased an Alpha Micro system four months ago:

AM-100	Software version 3.4
AM-200	W/ PerSci 277 dual floppy disk
AM-300	
T1810	Printer
SOROC	Terminal
Tectran	Digital cassette recorder

We have since interfaced a seven track 556 BPI tape transport, for purposes of communicating with our Honeywell H-200 mainframe.

We have learned how to use most of the system software, but there are some aspects where we have gotten no support:
For Example: SET PRG and the meaning of the error messages it returns, also one of your members indicated that Tech Tip # 15 should not be implemented.

We would appreciate any and all information that your group can pass to us, on the AM system and your organization

Sincerely,

S. C. Gilmore, Systems Analyst

CONSOLIDATED COMPUTER SYSTEMS

Software available:

1. PAYROLL -- A complete payroll package including check writing, journals, and tax reports.
2. ACCOUNT -- A General Accounting package for use by all types of businesses including accounting firms.
3. RECEIVE -- An Accounts Receivable package which interfaces with ACCOUNT. Includes full aging and billing.
4. PAYABLE -- An Accounts Payable package which interfaces with ACCOUNT. Includes check writing.
5. JOBCOST -- A complete Job Cost Accounting package for different types of industry/business. Interfaces completely with all above packages for automatic data entry. Forecasting subsystem for this package will be available soon, as well as a bidding subsystem for contract type businesses.
6. FINANCE -- Produces a user defined and user maintainable Financial Report package including Balance Sheet and Profit/Loss Report. Both reports may be easily formatted by the user prior to each printing. Interfaces with all above packages for automatic data entry.
7. CCSCRT -- A BASIC callable and stand alone sort package for the Alpha MicroSystems AM100 computer. This sort utilizes a very unique algorithm for both internal and external (disk based) sorts and is compatible with BASORT both in interface and timing. NO external temporary storage is required, thus allowing an entire Flexible Diskette to be sorted at one time.

CCSORT - SPECIFICATIONS

CCSORT is a Basic callable or stand-alone program that sorts contiguous random disk files in place without the use of any additional disk work space. The sort is quite rapid for files that take up half a floppy disk or less. It is somewhat more time consuming to sort files that occupy an entire floppy disk, but it is the only sort we know of with the ability to do so. Sort times are very competitive with BASORT/AMSORT, the package supplied by AlphaMicrosystems. CCSCRT is also smaller than BASORT/AMSORT and is totally re-entrant. The calling sequence for CCSORT is identical to that for BASORT for easy installation.

CCSCRT uses a unique sorting algorithm that completely eliminates merge phases for files that will not entirely fit into available memory. This coupled with an extremely rapid internal sorting algorithm and optimized input/output results in a reliable and fast sort.

Specifications:

Maximum record length..... 512 bytes
 Maximum records per file..... 32767 logical records
 Maximum sort keys supported.... 5 sets
 Key types supported..... Binary, floating point, string
 Temporary disk storage usage... NCNE
 Program size..... 1024 decimal bytes
 Minimum free memory required... 2048 decimal bytes (includes CCSCRT)

Typical sort times:

The following times are actual elapsed times to sort the specified files using an AM100 CPU with PerSci 277 disk in the AMOS format. User partition size is 40K bytes. The files are sorted on one four-byte binary key. The field I/O TIME is the elapsed time required just to read and write the file sequentially in BASIC. Compare these times to the total sort time. All times are in minutes and seconds.

Records To Sort	Record Length	Blocks In Use	Elapsed Times	I/O Time
-----	-----	-----	-----	-----
100	10	2	00:02	00:03
1000	10	20	00:26	00:32
100	32	7	00:03	00:04
1000	32	63	01:09	01:04
100	128	25	00:09	00:12
1000	128	250	02:58	01:54
100	256	50	00:16	00:15
1000	256	500	05:10	03:50
500	512	500	05:00	03:50

CCSORT is available from Consolidated Computer Systems,
 511 N. Akard St., Dallas, Tx. 75201

Computer Applications Research

ECONOMIC BATCH QUANTITY MODEL

Economic Batch Quantity Model (E.C.Q.M.) is a "User Friendly" system designed by C.A.R.

The object code is available from C.A.R. on a limited licence.

E.B.Q.M. PROVIDES THE OPTIMUM FREQUENCY AND BATCH SIZE SOLUTION TO INVENTORY ORDERING OR MANUFACTURING.

E.B.Q.M. utilizes a formula which encompasses all the variable factors involved in inventory or production planning. These factors are all inter-related in a fairly complex way. Some costs remain static and therefore are spread better over a higher inventory level (or larger production run). Some costs rise pro-rata. Some savings (discounts) occur in an infinitely variable (break-point) manner. E.B.Q.M. instantly finds the correct balance to produce the most economic ordering quantity and frequency. E.B.Q.M. also provides modelling facilities. This allows the User to instantly assess all the structural consequences of any alternative batch quantity.

INPUT FACTORS:

1. Annual demand for product.
2. Cost price per item.
3. Stock holding cost (warehousing, interest charges and general overheads, expressed as a percentage of the unit value).
4. Overhead cost per delivery (in manufacturing terms, machine setting-up, cost, transport).

After the input of parameters the E.B.Q.M. is instantly calculated and displayed with the following optimum values on one line:

1. Ordering frequency per annum.
2. Batch size.
3. Average stock holding.
4. Stock holding cost per annum.
5. Delivery cost per annum.
6. Total overhead cost.
7. Consolidated cost including cost saving benefits from the appropriate discount level.

MODELLING:

The optimum information remains displayed while the User is offered the modelling option.

From the further input of any specified batch quantity the same line of information as shown above is re-calculated and displayed for purposes of comparison with the E.B.Q.M.

SIGPC '79

First Annual Conference on Research and Development in Personal Computing
August 8-10, 1979 in Chicago at the Hyatt Regency O'Hare.
Sponsored by the Association for Computing Machinery (ACM)
Special Interest Group on Personal Computing (SIGPC)

SIGPC '79 will be held during Chicago Computer Visualization Week (August 6-10, 1979) along with the IEEE Pattern Recognition and Image Processing Conference (PRIP-79) and the ACM/SIGGRAPH Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '79). On August 10, SIGGRAPH will join us for joint sessions. A large trade show of personal computer and graphics equipment is planned to accompany an assortment of papers, panels, user group meetings, workshops, and person-to-person poster booths.

Papers are sought on research and development in personal computing, including, but not limited to, the following areas: computer architecture; data bases; human factors and interactive techniques; standards; innovative applications; input-output media; networks; operating systems; programming languages; software engineering; social and economic impacts; legal issues; portable systems; robotics.

CALL FOR PAPERS:

Papers submitted to technical sessions will be refereed by ACM criteria. To be accepted, a paper must contribute to the state of the art in personal computing, or be a tutorial paper exhibiting thorough knowledge of the area covered. Accepted papers will be published in a volume of proceedings to be distributed at the conference. The proceedings will be copyrighted under the new copyright law; each author must sign a release before publication. Selected papers of exceptional interest will be submitted for republication in the Communications of the ACM.

Papers should be submitted in quadruplicate. The length should not exceed 2000 words. The content should be accurate, understandable, and objective. The bibliography should give appropriate citations to related work.

General Chairmen:

Bob Gammill
Computer Science Division
Department of Mathematical Sciences
300 Minard Hall
North Dakota State University
Fargo, North Dakota 58102
(701) 237-8171

Tom DeFanti
Department of Information Engineering
University of Illinois at Chicago Circle
Box 4348 Chicago, Illinois 60680
(312) 996-2315

Program Committee:

Dennis Allison, Stanford University
Robert Chew, Tektronix
Mike Faiman, University of Illinois
Dick Hackathorn, Wharton School
Gary Kildall, Digital Research
Susan Nycum, Chickering and Gregory
Harry Saal, IBM
Larry Tesler, Xerox (chairman)

Deadlines:

January 5, 1979: 300 word abstract
February 9, 1979: 4 copies of first draft & figures
April 16, 1979: notification to authors
May 4, 1979: final draft typed on special forms

Please send the form below and an abstract by January 5, 1979 to:

Larry Tesler
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, Ca 94304
(415) 494-4355

SIGPC '79:

I intend to submit a paper on the topic: _____

I propose to organize a panel discussion about: _____

I propose to organize a users group meeting for: _____

Name: _____

Telephone: _____

Affiliation: _____

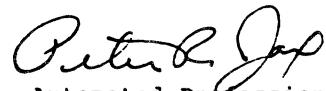
Address: _____

City: _____ State: _____ Zip: _____

The following is my short paragraph of changes and enhancements to be suggested to Alpha Micro. They are not in priority order at this time.

1. More protection from unauthorized access from another PPN.
2. High quality service from a national organization.
3. Documentation on I/O areas of AMOS for customizing on client systems.
4. FORTRAN
5. COBOL
6. APL
7. PASCAL run versions which run under AMOS and chain to and from BASIC and FORTRAN programs.
8. Add /D option to PRINT command to delete file after printing.
9. Allow options for multiple copies and special forms on PRINT
10. Documentation on "Running your own monitor".
11. Faster backup facilities.
12. Tape availability.
13. Larger record size than 512.
14. Variable record sizes.
15. File access of sequential files in APPEND and UPDATE mode.
16. Function definition in BASIC.
17. MATrix operation in BASIC.
18. BASIC to BASIC subroutine calls or overlay.
19. Prevention of crash under "OUT OF MEMORY".
20. In BASIC default SAVE to save under last LOADED name and add name field to NEW command to prevent SAVEing over another program by mistake.
21. Allow looping in command files.
22. Full Screen editor (I heard a rumor about VUE).
23. One simple editing feature in BASIC like CHANGE or ALTER (I'd be glad to offer about 10 specific suggestions).
24. Request PASSWORD every time you LOG even if PASSWORD is null.
25. Print Version number and release date on inside cover of all manuals.
26. Don't echo input until it is used so it doesn't destroy the visual fidelity of a user screen.
27. Make source for Accounting subroutines available for customizing.
28. More than one SPOOLED printer on system.
29. Remove default of all from ERASE. It's too dangerous.

Peter R. Jax



Automated Professional
Systems Inc

University Computer Club
University Software Society
Babbage Society
Guild of Undergraduates
University of W.A.
Nedlands Western Australia 6009.

AlphaMicro Users' Society: Membership Application

Name: Dean Lawrence Elsnor
Company: Babbage Society Inc. (Pres)
Address: C/O University Computer Club
Guild of Undergraduates
University of W.A.
Australia 6009.

Business; phone: Australia; 09-380-2297
Hardware: Own AM100, AM200, AM300
Also: networked with TWO cybers and TWO DEC-10s as front ends.
Also: sundry 2650s, 8080s, Z80s, soft&hard sector floppys ...

Grade: Individual user(s)
Have: Terminal drivers;
Baudot
Generalised TDY that has editing lingo built in
(insert, delete, search, copy...)
Certain selectrics

Utilities:
Disk and memory diagnostics
Disk file managers (much better than COPY)
Text formatter /Word processor (not yet released)
Network software: protocols for DEC, IBM, CDC.

(We are in fact a bunch of amateurs... but we have been going since 1974)

please find enclosed cheque for \$25 U.S.
please send back issues of AMUS newsletter ~~XXXXX~~ from #7.
when joined, I can send you quite a few bug fixes to trmser, pseudo etc.

Dean Elsnor

Dean ELSNER
16 Viking Road
Dalkeith 6009
Western Australia
(09)386-6436

6-Oct-78

OCT 17 1978



NORTH AMERICA TITLE COMPANY

107 Welch • Houston, Texas 77006 • 713 526-2725
4830 Loch Lomond • Houston, Texas 77096 • 713 667-3021

November 30, 1978

NEED TO EXCHANGE DATA BETWEEN THE OUTSIDE WORLD AND YOUR ALPHA-MICRO?

We have a package to allow you to exchange data via the IBM 3747 series machines and the Alpha-Micro.

Capabilities:

1. Go from 9-track 800BPI tape to floppy disks via the IBM 3747(or equivalent) then to your Alpha-Micro.
2. Dump data or program files from your AMOS structured devices to an IBM 3747 compatible floppy disk and then to tape.

How it works:

Three assembly language routines called from a user basic program form the heart of the system. These allow you to open a IBM compatible floppy for input/output, read off a file to an AMOS file, or write a file of your choice onto the floppy. ASCII to EBCDIC conversions are perform automatically for you. These subroutines can be used in conjunction with a custom basic program to fit almost any needs.

Other programs in the Package:.

OUT.BAS	Sample general utility program for writing files from AMOS to IBM 3747 format.
INPUT.BAS	Sample general utility program for reading files from IBM 3747 format to an AMOS file.
IGMDMP.BAS	Utilitiy program used to examine data on an IBM 3747 floppy.
IMGHEX.BAS	Utility program used to examine data on an IBM 3747 floppy in Hex with out EBCDIC to ASCII conversion
DUMP.BAS	Utility program to examine an random AMOS file.
?????.CMD	Sample command files to use the above programs and by which to model your own.

COST: \$400.00

TO ORDER OR FOR ADDITIONAL INFORMATION CONTACT:

Eugene C. Platt
10333 Northwest Freeway
#518
Houston Texas 77092
713/681-6949

the data bank

db

70 west palmetto drive
paradena, california 91105
(213) 795-8611

November 10, 1978

Jim Taylor
Alpha Micro User's Society
P.O. Box 1723
Boulder, Colorado 80306

Dear Jim:

Enclosed is a copy of our User's Manual for AMAR (Alpha Micro Address Retention).

We developed AMAR in April for our own use. The Data Bank does a lot of mailing list maintenance for accountants and bookkeeping services and we needed a program that would allow us to manipulate many small mailing lists quickly and efficiently utilizing only one floppy disk.

AMAR stores first name, last name, 2 address lines, city, state, zip, telephone number and an optional comment field for each record. The input data is edited for length and format prior to storage and can be easily retrieved on the CRT or in printed format. There are twenty programs in the system, giving the user the flexibility to do just about anything he wants with the data stored. The user can access any file (or any record in any file) on request. AMAR monitors the number of records in each file and the number of files residing on the disk and keeps the user apprised of these totals. Galley sheets (name & address rosters) may be printed or displayed on the CRT at any time and mailing labels may be printed in 3, 4 or 5 line format. The sort module allows the user to sort on any field in the record (up to 3 sort keys) or he may optionally choose to sort in any one of 7 standard formats. The entire system is menu-oriented with CRT screen formatting features that make AMAR very easy to use.

We are making the object code available to AM-100 users for \$50 including the User's Manual. (The user's manual may be ordered for \$4.)

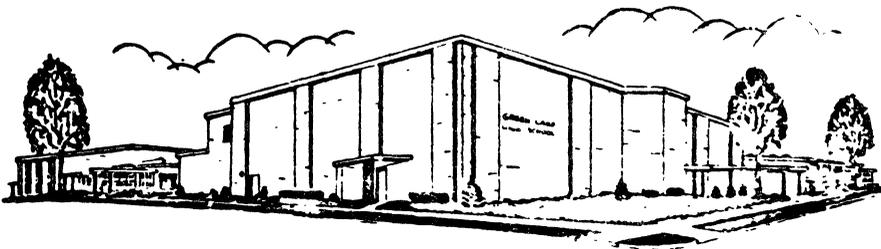
The source code is available to AM-100 dealers only on a limited license basis.

Sincerely,



Tom Herod

Encls.



GREEN LAKE PUBLIC SCHOOLS

GREEN LAKE, WISCONSIN 54941

PHILIP BARANOWSKI
SUPERINTENDENT

Oct. 31, 1978

Mr. Jim Taylor, President
Alpha Micro Users Soc.
Box 1723
Boulder, Colo. 80306

Mr. Taylor:

NOV 15 1978

Last July, the Green Lake School District purchased an AM-100 computer system, including a dual floppy drive, 64K memory, 3-CRT'S, and a hard copy printer.

We are an educational institution, k-12, and are developing applications programs, in BASIC, for the above system. Many of the applications are modifications of programs previously written for a DEC PDP-8I.

Currently, we have running:

- 1) Attendance Accounting system
- 2) Grade reporting systems
- 3) Student Directory and Mailer
- 4) A variety of Math and Spelling drills
- 5) Plotters, for algebraic equations

Future activities will include;

- 1) District Accounting
- 2) Payroll
- 3) District Census
- 4) Other administrative and academic uses.

I would be very interested in contributing to and receiving information from a Users Society such as the one I have seen described in recent releases. Please include our name and send information concerning membership, etc.

Your cooperation is greatly appreciated.

Sincerely,
THOMAS J. Oyster

Coordinator of Computer Activities, Green Lake Schools

J. GIBBONS ZERATSKY, PRESIDENT

ANN SANDLEBACK, CLERK

HAROLD BIRKHOLOZ, TREASURER

ARNOLD ALVIN

GEORGE MABEE

SALLY McNEILY

GERALD KAUTZER

ALPHA MICRO USERS SOCIETY Membership form

Please fill out as much information as possible.

Name _____ Company _____

Address _____ City _____ State _____ Zip _____

Business Phone _____ Home Phone _____

Circle one: Own Lease Thinking

Check all applicable: Dealer _____ OEM _____ User: Corporate _____ Individual _____

Describe equipment: _____

AMUS may use my name for mailing lists

Make checks payable to AMUS

Annual dues are \$25.00 per member.

ALPHA MICRO USERS SOCIETY
c/o COMMUNITY FREE SCHOOL
PO BOX 1724, BOULDER, COLO. 80306

* NON-PROFIT ORGANIZATION *
* U.S. POSTAGE PAID *
* PERMIT 490 *
* BOULDER, COLORADO *
* 80306 *
